УДК 004.89:004.93

О.В. Бармак, Ю.В. Крак, К.С. Кручинін
Інститут кібернетики НАН України, м. Київ
Україна, 03680, просп. Академіка Глушкова, 40, м. Київ, МСП

Стійкі ознаки зображення для ідентифікації конфігурації руки в українській жестовій мові

O.V. Barmak, Iu.V. Krak, K.V. Krychinin
V.M. Glushkov Institute of Cybernetic of NAS of Ukraine
Ukraine, 03680, Kiev, Glushkova ave., 40

Stable Features of Images for Identification of Hand Configuration of Ukrainian Sign Language

А.В. Бармак, Ю.В. Крак, К.С. Кручинин
Институт кибернетики НАН Украины, г. Киев
Украина, 03680, просп. Акад. Глушкова, 40, г. Киев, МСП

Устойчивые признаки изображения для идентификации конфигурации руки в украинском жестовом языке

В статті розглядається задача ідентифікації на зображенні конфігурації руки для української жестової мови. Запропоновано для розв'язку цієї задачі використовувати метод Speeded Up Robust Features (SURF), оскільки він є одним з найбільш ефективних і швидких сучасних алгоритмів. Отримані експериментальні результати показали досить високу ефективність методу для ідентифікації дактильних знаків.
Ключові слова: розпізнавання жестів, ідентифікація жесту.

The problem of identification of hand configuration on an image for Ukrainian Sign language is studied in this paper. To solve the problem a method Speeded Up Robust Features (SURF) is suggested to be used since it is one of the most effective and rapid of the present-day algorithms. The obtained experimental results showed a fairly high efficiency of the method for identification of dactyl signs.
Key words: gesture recognition, identification of gesture.

В статье рассматривается задача идентификации на изображении конфигурации руки для украинского жестового языка. Предложено для решения этой задачи использовать метод Speeded Up Robust Features (SURF), так как он есть одним из наиболее эффективных и быстрых современных алгоритмов. Полученные экспериментальные результаты показали достаточно высокую эффективность метода для идентификации дактильных знаков.
Ключевые слова: распознавание жестов, идентификация жеста.

Вступ

Задача розпізнавання образів до цього часу не вирішена в повному обсязі. Проте, в рамках істотних обмежень, є методи, що дозволяють наблизитись до її вирішення. Серед різних споріднених методів заслуговує на увагу метод Speeded Up Robust Features (SURF), оскільки він є одним з найбільш ефективних і швидких сучасних алгоритмів [1].

Розглядаючи задачу ідентифікації на зображенні конфігурації руки для української жестової мови, можна виділити ряд моментів, врахування яких важливе для успішного розв'язку: 1) масштаб – зображення мають різний масштаб, тобто предмети, які сприй-

маються як однакові, насправді займають різну площу на різних зображеннях; 2) локалізація – об’єкт може знаходитися в різних місцях зображення; 3) фон і шуми – предмет, який сприймається як щось окреме, на зображенні ніяк не виділений, і знаходиться на тлі інших предметів; 4) проекція, обертання і кут зору – зображення є лише двовимірною проекцією тривимірного світу, тому поворот об'єкта і зміна кута зору загалом можуть кардинальним чином впливати на його двовимірну проекцію.

Метою даної роботи є визначення з двох зображень (одне з них вважається зразком і містить можливі конфігурації руки при показі дактильної абетки, інше вважається сценою і містить руку з конфігурацією конкретного знака) факту наявності конкретного зразка на сцені і його локалізації. При цьому потрібно врахувати означені вище моменти.

Метод пошуку та опису особливих точок зображення SURF

Основна ідея методу SURF полягає у виділенні на зразку певних ключових точок і невеликих ділянок навколо них. Ключовою точкою вважається така точка, яка має певні ознаки, які істотно відрізняють її від основної маси точок. Наприклад, це можуть бути краї ліній, невеликі кола, різкі перепади освітленості, кути тощо. Припускаючи, що ключові точки присутні на зразку завжди, можемо пошук зразка звести до пошуку на сцені ключових точок зразка. А оскільки ключові точки сильно відрізняються від основної маси точок, то їх кількість буде істотно менше, ніж загальна кількість точок зразка. У цілому, принцип вибору ключових точок не важливий, головне що б їх було не дуже багато і вони були присутні на зображенні зразка завжди.

Невеликі ділянки навколо ключових точок потрібні для наступного. Чим менше ділянка, тим менше на неї впливають великомасштабні спотворення. Так, якщо об’єкт у цілому, піддається ефекту перспективи (тобто ближній край об’єкта має більший видимий розмір, ніж далекий), то для малої ділянки явищем перспективи можна знехтувати і замінити на зміну масштабу. Аналогічно, невеликий поворот об’єкта навколо деякої осі може сильно змінити картинку об’єкта в цілому, але малі ділянки зміняться трохи. Крім того, якщо частина об’єкта виходить за край зображення або закрита, то невеликі ділянки навколо частини ключових точок будуть видні повністю, що також дозволяє їх легше ідентифікувати. А ще, якщо малі області лежать цілком усередині шуканого об’єкта, то на них не мають жодного впливу об’єкти фону. З іншого боку, ділянка навколо ключової точки не повинна бути занадто малою. Дуже малі ділянки несуть занадто мало інформації про зображення й з більшою ймовірністю можуть випадково співпадати між собою.

Метод SURF вирішує два завдання – пошук особливих точок зображення – створення їх дескрипторів, інваріантних до масштабу і обертання. Це означає, що опис ключової точки буде однаковий, навіть якщо зразок змінить розмір і буде повернений (тут і далі ми будемо говорити тільки про обертання у площині зображення). Крім того, сам пошук ключових точок теж повинен мати інваріантність так, щоб повернутий об’єкт сцени мав той же набір ключових точок, що і зразок.

Метод шукає особливі точки за допомогою матриці Гессе. Детермінант матриці Гессе (гессіан) досягає екстремуму в точках максимальної зміни градієнта яскравості. Він добре детектує плями, кути і краї ліній. Гессіан інваріантний щодо обертання, але не інваріантний до масштабування. Тому SURF використовує різномасштабні фільтри для знаходження гессіанів. Для кожної ключової точки обраховується напрямок максимальної зміни яскравості (градієнт) і масштаб, взятий з масштабного коефіцієнта матриці Гессе. Градієнт в точці обчислюється за допомогою фільтрів Хаара.

Після знаходження ключових точок, SURF формує їх дескриптори. Дескриптор – це набір з 64 (або 128) чисел для кожної ключової точки. Ці числа відображають флуктуації градієнта навколо ключової точки. Оскільки ключова точка є максимумом гессіана, то це гарантує, що в околиці точки мають бути ділянки з різними градієнтами. Таким чином, забезпечується дисперсія (відмінність) дескрипторів для різних ключових точок.

Флуктуації градієнта околиць ключової точки обраховуються щодо напрямку градієнта навколо точки в цілому (по всій околиці ключової точки). Таким чином, досягається інваріантність дескриптора щодо обертання. Розмір області на якій обраховується дескриптор, визначається масштабом матриці Гессе, що забезпечує інваріантність щодо масштабу. Флуктуації градієнта також обраховуються з допомогою фільтра Хаара.

Для ефективного обчислення фільтрів Гессе і Хаара – використовується інтегральне подання зображень. Інтегральне подання є матрицею, розмірність якої співпадає з розмірністю вихідного зображення, а елементи обраховуються за формулою:

$$II(x,y) = \sum_{i=0,j=0}^{i\le x, j\le y} I(i,j), \tag{1}$$

де $I(i,j)$ – яскравість пікселів вхідного зображення.

Маючи інтегральну матрицю можна дуже швидко обчислювати суму яскравостей пікселів довільних прямокутних областей зображення, за формулою:

$$SumOfRect(ABCD)=II(A)+II(B)+II(C)+II(D), \tag{2}$$

де $ABCD$ – прямокутник, який нас цікавить.

Виявлення особливих точок у SURF базується на обчисленні детермінанта гессіана. Матриця Гессе для двовимірної функції та її детермінант визначається наступним чином:

$$H(f(x,y)) = \begin{bmatrix} \partial^2 f/\partial x^2 & \partial^2 f/\partial x\partial y \\ \partial^2 f/\partial x\partial y & \partial^2 f/\partial y^2 \end{bmatrix}, \quad \det(H) = \frac{\partial^2 f}{\partial x^2}\frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial x\partial y}\right)^2. \tag{3}$$

Значення гессіана використовується для знаходження локального мінімуму або максимуму яскравості зображення. У цих точках значення гессіана досягає екстремуму.

Зазвичай, обчислення матриці Гессе зводиться до знаходження лапласіана Гауссіана. По суті елементи матриці Гессе обчислюються як згортка (сума добутків) пікселів зображення на спеціальні фільтри (рис. 1). Однак SURF не використовує лапласіан гауссіана в тому вигляді, який зображений на рис. 1. По-перше, за твердженням авторів методу [1], дискретизований лапласіан гауссіана має досить великий розкид значень детермінанта при обертанні зразка (нагадаємо, що в ідеалі гессіан має бути інваріантним до обертання). Особливо детермінант «просідає» в районі повороту на 45 градусів. По-друге, і це головне, фільтр для лапласіана гауссіана має неперервний характер. Майже всі пікселі фільтра мають різні величини яскравості. А це не дозволяє ефективно використовувати такий потужний механізм розрахунку, як інтегральну матрицю зображення. Тому SURF використовує бінарізовану апроксимацію лапласіана гауссіана (Fast-Hessian) (рис. 2):

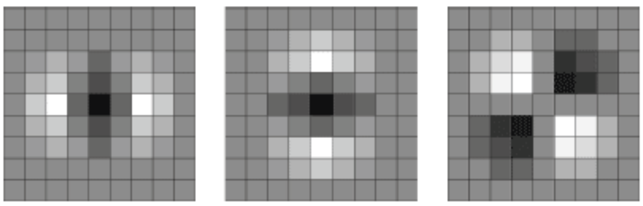


Рисунок 1 – Фільтр для звичайного обчислення матриці Гесе

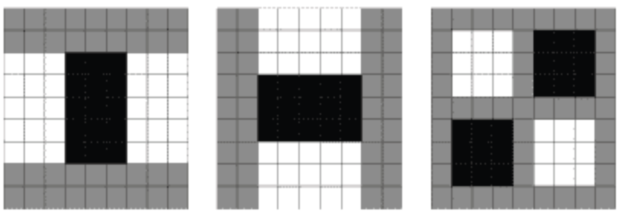


Рисунок 2 – Фільтр для обчислення Fast-Hessian

Цей фільтр більш стійкий до обертання і його можна ефективно обчислити за допомогою інтегральної матриці. Таким чином, в SURF гессіан обчислюється так:

$$\det\left(H_{approx}\right) = D_{xx}D_{yy} - \left(0.9D_{xy}\right)^2, \quad (4)$$

де D_{xx}, D_{yy}, D_{xy} – згортки по фільтрах, зображених на рис. 2. Коефіцієнт *0.9* має теоретичне обґрунтування, і коригує наближений характер обчислень.

Отже, для знаходження особливих точок, SURF пробігає по пікселях зображення і шукає максимум гессіана. У методі задається порогове значення гессіана. Якщо обчислене значення для пікселя вище порога – піксель розглядається як кандидат на ключову точку. Оскільки гессіан є похідною, і залежить тільки від перепаду яскравості, а не від абсолютного її рівня, то він інваріантний по відношенню до зсуву яскравості зображення. Таким чином, зміна рівня освітлення зразка не впливає на виявлення ключових точок. Крім того, властивості гессіана такі, що він досягає максимуму, як в точці білої плями на чорному тлі, так і чорної плями на білому тлі. Таким чином, метод виявляє і темні, і світлі особливості зображення (рис. 3).

Як вже зазначалося, гессіан не інваріантний відносно масштабу. Це означає, що для одного і того ж піксела, гессіан може змінюватися при зміні масштабу фільтра. Вирішення цієї проблеми тільки одне - перебирати різні масштаби фільтрів і по черзі їх застосовувати до даного піксела. З міркувань симетрії і дискретизації, розмір фільтра Fast-Hessian не може приймати довільні значення. Допустимі розміри цього фільтра такі (починаючи з мінімального): 9, 15, 21, 27 і так далі, з кроком 6. Однак, на практиці, поступово збільшувати розмір фільтра на 6 не доцільно, тому-що для великих масштабів крок 6 виявляється занадто дрібним, а фільтри – надмірними. Тому (і з деяких інших причин), SURF розбиває безліч масштабів на так звані октави. Кожна октава покриває певний інтервал масштабів і має свій характерний розмір фільтра. При цьому якщо б на октаву припадав лише один фільтр, це було б занадто грубим наближенням. Крім того, не можна було б знайти локальний максимум гессіана серед різних масштабів у різних октавах. Адже одна і та ж точка може мати декілька локальних максимумів гессіана в різних масштабах.

Якщо шукатимемо максимум серед усіх гессіанів по всіх масштабах, то знайдемо тільки один з максимумів, у той час як їх може бути декілька. Один – в одному масштабі, інший – в іншому. Виходячи з перерахованого, октава містить не один фільтр, а чотири фільтри, які добре покривають характерний масштаб октави (рис. 4).

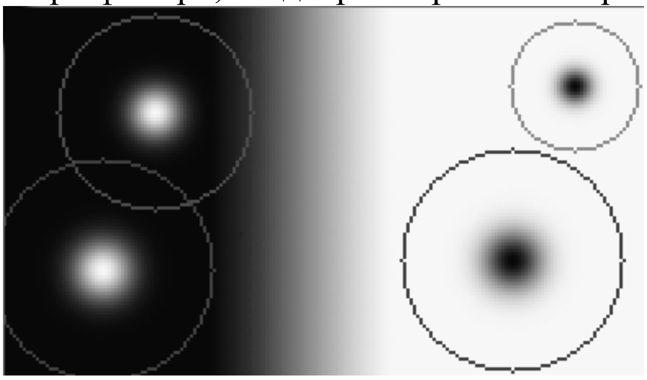


Рисунок 3 – Метод розпізнає як світлі точки на темному фоні, так і темні точки на світлому фоні

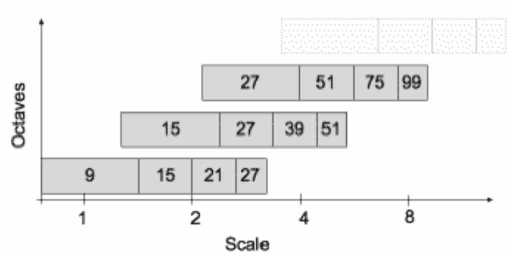


Рисунок 4 – Перші три октави методу SURF. Цифри в прямокутниках показують розмір фільтра Fast-Hessian. Логарифмічна шкала знизу - показує масштаби, що покриваються октавами

Крок розміру фільтра у першій октаві складає 6, у другій – 12, у третій – 24 тощо. Як бачимо, октави значно перекриваються одна одною. Це збільшує надійність знаходження локальних максимумів. Теоретично, масштаби нескінченні, проте в реальних зображеннях, вони цілком скінченні і основна маса зосереджена в інтервалі від 1 до 10 (за даними [1]). Для покриття цього діапазону достатньо чотирьох октав. Плюс додається одна або дві октави для покриття більших масштабів. Разом використовується 5 – 6 октав. Теоретично цього досить для покриття всіх масштабів на зображенні 1024×768пікселів.

Для знаходження локального максимуму гессіана використовується метод сусідніх точок 3×3×3. Його сенс зрозумілий з рис. 5.

Для інваріантності обчислення дескрипторів особливої точки потрібно визначити переважаючу орієнтацію перепадів яскравості в особливій точці. Це поняття близьке до поняття градієнта, але SURF використовує трохи інший алгоритм знаходження вектора орієнтації. Спочатку, обчислюються точкові градієнти в пікселах, сусідніх з особливою точкою. Для розгляду беруться пікселі в кола радіусу *6s* навколо особливої точки, де *s* - масштаб особливої точки. Для першої октави беруться точки з околу радіусу 12. Для обчислення градієнта, використовується фільтр Хаара, вигляд яких зображено на рис. 6. Розмір фільтра береться рівним *4s*.

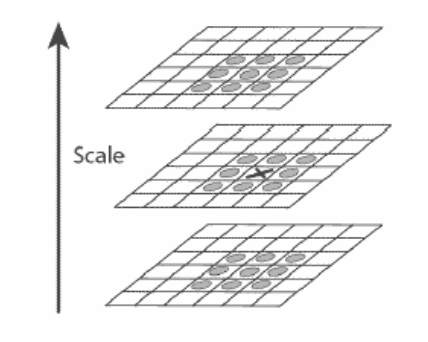


Рисунок 5 – Піксел, позначений хрестиком вважається локальним максимумом, якщо його гессіан більше ніж у будь-якого його сусіда в його масштабі, а також більше будь-якого із сусідів масштабом менше і масштабом більше (всього 26 сусідів)

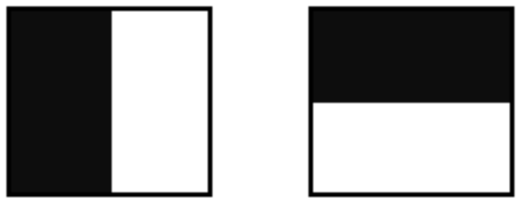


Рисунок 6 – Фільтри Хаара. Чорні області мають значення -*1*, білі +*1*

Фільтри Хаара дають точкове значення перепаду яскравості по осям *X* і *Y* відповідно. Оскільки фільтри Хаара мають прямокутну форму, їх значення легко обчислюються за допомогою інтегральної матриці. Для розрахунку одного фільтра довільного розміру потрібно всього 6 операцій. Значення вейвлета Хаара *dX* і *dY* для кожної точки множаться на вагу і запам’ятовуються у масиві. Вага визначається як значення гауссіана з центром в особливій точці і сигмою рівною *2s*. Зважування на гауссіані необхідно для відсікання випадкових перешкод на далеких від особливої точки відстаннях. Далі всі знайдені значення *dX* і *dY* умовно наносяться у вигляді точок на площину, як показано на рис. 7.

Наступним кроком береться кутове вікно розміром *π/3*, і обертається навколо центру координат. Вибирається таке положення вікна, при якому довжина сумарного вектора для точок, що потрапили у вікно – максимальна. Обчислений таким чином вектор нормується і приймається як пріоритетний напрямок в області особливої точки. Маніпуляції з вікном потрібні для зменшення впливу точок-шумів. На рис. 8 приведений приклад градієнта при ідеальному краї і при краї з шумами. Як бачимо, шум дає додаткові градієнти в напрямках, які не збігаються з напрямом основного

градієнта. Використання вікна дозволяє відсікти такі точки-шуми і більш точно обчислити справжній градієнт. Відзначимо, що не завжди потрібна інваріантність дескрипторів щодо обертання. Метод SURF має модифікацію в якій орієнтація особливих точок не розраховується. Така модифікація дозволяє надійно ідентифікувати точки, повернені не більше ніж на ± 15 градусів.

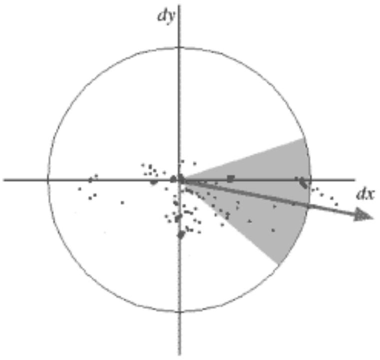


Рисунок 7 – Знайдені градієнти у вигляді точок у просторі *dXdY*

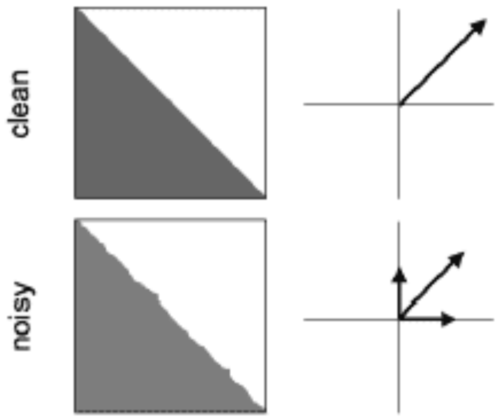


Рисунок 8 – Приклад градієнта при ідеальному краї і при краї з шумами

Дескриптор – це масив з 64 (у розширеній версії 128) чисел, що дозволяють ідентифікувати особливу точку. Дескриптори однієї і тієї ж особливої точки на зразку і на сцені мають приблизно збігатися. Метод розрахунку дескриптора такий, що він не залежить від обертання і масштабу. Для обчислення дескриптора, навколо особливої точки формується прямокутна область, що має розмір *20s*, де *s* – масштаб в якому була знайдена особлива точка. Для першої октави, область має розмір 40×40 пікселів. Квадрат орієнтується уздовж пріоритетного напрямку, обчисленого для особливої точки. Дескриптор розраховується як опис градієнта для 16 квадрантів навколо особливої точки. Цей квадрат розбивається на 16 дрібніших квадрантів (рис. 9). У кожному квадраті береться регулярна сітка 5×5 і для точки сітки шукається градієнт, за допомогою фільтра Хаара. Розмір фільтра Хаара береться рівним *2s*, і для першої октави становить 4×4. Слід зазначити, що при розрахунку фільтра Хаара, зображення не повертається, фільтр обчислюється в звичайних координатах зображення, а отримані координати градієнта (*dX, dY*) повертаються на кут, відповідний орієнтації квадрата. Разом, для обчислення дескриптора особливої точки, потрібно обчислити 25 фільтрів Хаара в кожному з 16 квадрантів. Разом 400 фільтрів Хаара. Враховуючи що на фільтр потрібно 6 операцій, виходить, що дескриптор обійдеться мінімум в 2400 операцій. Після знаходження 25 точкових градієнта квадранта, обчислюються чотири величини, які власне і є компонентами дескриптора: $\sum dX$, $\sum |dX|$, $\sum dY$, $\sum |dY|$. Дві з них є просто сумарний градієнт по квадранту, а дві інших – сума модулів точкових градієнтів. На рис. 10 показана поведінка цих величин для різних ділянок зображення. Для рівномірних ділянок – всі значення близькі до нуля. Для вертикальних полос, що повторюються – всі величини, крім другої близькі до нуля. При збільшені яскравості у напрямку осі *X*, дві перші компоненти мають більші значення.

Чотири компоненти на кожний квадрант і 16 квадрантів дають 64 компоненти дескриптора для всієї області особливої точки. При занесенні в масив, значення дескрипторів зважуються на гауссіан з центром в особливій точці і з сигмою *3.3s*. Це потрібно для більшої стійкості дескриптора до шумів у віддалених від особливої

точки областях. Окрім дескриптора, для опису точки використовується знак сліду матриці Гесе, тобто величина *sign (Dxx + Dyy)*. Для світлих точок на темному тлі, слід негативний, для темних точок на світлому фоні – позитивний. Таким чином, SURF розрізняє світлі і темні плями.

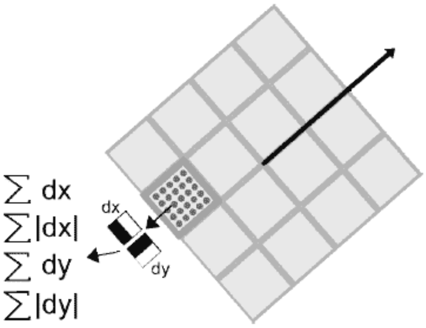


Рисунок 9 – Розбивка квадрата на 16 квадрантів

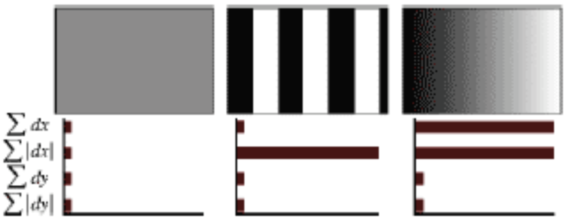


Рисунок 10 – Поведінка дескриптора для різних зображень

Реалізація методу для ідентифікації знаків дактильної абетки

Метод SURF реалізований у відкритих бібліотеках [3,4]. Для проведення експериментів створено додаток, що використовує відповідні процедури із бібліотеки OpenCV [4]. Застосовувалась наступна алгоритмічна послідовність:

1) створювався об’єкт **`surfParam`** класу **`MCvSURFParams`** з порогом для витягування особливостей тільки з великим значенням гессіану (прийнятне значення складає приблизно 300-500 і може залежати від середнього локального контрасту і чіткості зображення) з базисним дескриптором по 64 елемента;

2) створювався об’єкт-зразок зображення, що містив еталони дактильної абетки у виконанні конкретної людини;

3) витягувалися особливості (**`modelFeatures`**) із об’єкта-зразка за допомогою методу **`ExtractSURF`** з параметром **`surfParam`**;

4) створювався об’єкт **`tracker`** класу **`SURFTracker`** з параметром **`modelFeatures`** для відслідкування об’єкта;

5) створювався об’єкт-сцена та витягувались особливості (**`imageFeatures`**) методу **`ExtractSURF`** з параметром **`surfParam`**;

6) узгоджувались особливості об’єкта та сцени (**`matchedFeatures`**) за допомогою метода `MatchFeature` об’єкта **`tracker`**;

7) фільтрувалися узгоджені особливості (відкидались ті відповідності яких не унікальні);

8) відкидалися узгоджені особливості, масштаб і обертання яких не задовольняло масштабу і обертанню для більшості об’єктів.

Висновки

Проведені експерименти показали досить високу ефективність методу для ідентифікації дактильних знаків української жестової мови. Верхня частина зображення (рис. 11) містить зразки конфігурацій руки при показі дактильної абетки. Нижня частина зображення містить сцену з людиною, що показує певну букву. Алгоритм стало розпізнає всі дактилеми (букви).

Подальші дослідження направленні на вдосконалення та поширення запропонованого підходу на жести української жестової мови.



Рисунок 11 – Ідентифікація дактильних знаків української жестової мови за допомогою методу SURF

Литература

1. Bay H. SURF: Speeded Up Robust Features / H. Bay, A. Ess, T. Tuytelaars, L. Van Gool // Computer Vision and Image Understanding (CVIU). – 2008. – Vol. 110, No. 3 – P. 346-359.
2. http://www.chrisevansdev.com/computer-vision-opensurf.html The OpenSURF Computer Vision Library.
3. http://sourceforge.net/projects/opencvlibrary/ Open Computer Vision Library.

O.V. Barmak, Iu.V. Krak, K.V. Krychinin

Stable Features of Images for Identification of Hand Configuration of Ukrainian Sign Language

The problem of pattern recognition isn’t solved in full yet. However, there are methods that allow approach to its solution under significant restrictions. Among the various related methods, the method Speeded Up Robust Features (SURF) deserves attention because it is one of the most effective and rapid of the modern algorithms.

The problem of identification of a hand configuration on an image for Ukrainian Sign Language is studied in this paper. Mentioned method is proposed for solving this problem. Method SURF takes into account some important points for successful solution: 1) independent from image scale, 2) independent from localization of an object that we are interested in on an image, 3) independent from a background and noise, 4) independent from a projection, rotation and angle of view.

The purpose of this study is to determine from two images the existence of a specific sample on the scene and its localization (one image is the sample and contains the possible configurations of the hand that shows dactyl ABC, the other is the scene that contains a hand in a configuration of a particular sign). Thus the above points are necessary to be taken into account.

The experiments showed sufficiently high efficiency of the method of identification of a dactyl sings of Ukrainian Sign Language. The algorithm successfully detects all dactyls (letters).

Further studies are direct at improving and extending suggested approach to gestures of Ukrainian Sign Language.

Стаття надійшла до редакції 10.05.2012.